数理解析研究所講究録
666巻 1988年 71-80

2次元マーカオートマトンのある性質
--- 3方向チューリング機械による模倣 ---

山口大学工短部　伊藤　暁　(Akira Ito)
山口大学工学部　井上克司　(Katsushi Inoue)、　高浪五男　(Itsuo Takanami)

1. Introduction and Preliminaries. We denote a two-dimensional deterministic (nondeterministic) one-marker automaton by "2-DM_1" ("2-NM_1"), and a three-way two-dimensional deterministic (nondeterministic) Turing machine by "TR2-DTM" ("TR2-NTM"). In this paper, we show that the necessary and sufficient space for TR2-NTM's to simulate 2-DM_1's (2-NM_1's) is $n \log n$ (n^2), and the necessary and sufficient space for TR2-DTM's to simulate 2-DM_1's (2-NM_1's) is $2^{O(n \log n)}$ ($2^{O(n^2)}$), where n is the number of columns of rectangular input tapes.

In this paper, the detailed definitions of two-dimensional marker automata and (space-bounded) three-way two-dimensional Turing machines are omitted. If necessary, refer to [1,2].

Definition 1. Let Σ be a finite set of symbols. A *two-dimensional tape* over Σ is a two-dimensional rectangular array of elements of Σ.

The set of all two-dimensional tapes over Σ is denoted by $\Sigma^{(2)}$.

For a tape $x \in \Sigma^{(2)}$, we let $\ell_1(x)$ be the number of rows of x and $\ell_2(x)$ be the number of columns of x. If $1 \le i \le \ell_1(x)$ and $1 \le j \le \ell_2(x)$, we let x(i,j) denote the symbol in x with coordinates (i,j). Furthermore, we define

$$x[(i,j),(i',j')],$$

when $1 \le i \le i' \le \ell_1(x)$ and $1 \le j \le j' \le \ell_2(x)$, as the two-dimensional tape z satisfying the following:

(i) $\ell_1(z)=i'-i+1$ and $\ell_2(z)=j'-j+1$,

(ii) for each k,r $[1 \le k \le \ell_1(z), 1 \le r \le \ell_2(z)]$, $z(k,r)=x(k+i-1,r+j-1)$.

When a two-dimensional tape x is given to any two-dimensional automaton as an input, x is surrounded by the boundary symbol "#"s.

Definition 2. Let x be in $\Sigma^{(2)}$ and $\ell_2(x)=n$. When $\ell_1(x)$ is divided by n, we call

$$x[((j-1)n+1,1),(jn,n)]$$

an *n-block of x,* for each $j(1 \le j \le \ell_1(x)/n)$.

Definition 3. For any two-dimensional automaton M with input alphabet Σ, define

$T(M)=\{x \in \Sigma^{(2)} \mid M \text{ accepts } x\}$.

Furthermore, define

$\mathcal{L}[\text{2-DM}_1]=\{T \mid T=T(M) \text{ for some 2-DM}_1 \text{ M}\}$ and

$\mathcal{L}[\text{2-NM}_1]=\{T \mid T=T(M) \text{ for some 2-NM}_1 \text{ M}\}$.

We similarly define $\mathcal{L}[\text{TR2-DTM}(L(m,n))]$ ($\mathcal{L}[\text{TR2-NTM}(L(m,n))]$) as the class of sets accepted by L(m,n) space-bounded TR2-DTMs (TR2-NTMs).

By using an ordinary technique, We can easily show that the following theorem holds.

Theorem 1. *For any function $L(m,n) \ge \log n$,*

$$\mathcal{L}[\text{TR2-NTM}(L(m,n))] \subseteq \mathcal{L}[\text{TR2-DTM}(2^{O(L(m,n))})].$$

2. Sufficient Space.

In this section, we investigate the sufficient space for three-way Turing machines to simulate 1-marker automata.

We first show that n log n space is sufficient for TR2-NTM's to simulate 2-DM_1's.

Theorem 2. *$\mathcal{L}$[2-DM_1] $\subseteq$ $\mathcal{L}$[TR2-NTM(n log n)].*

Proof. Suppose that a 2-DM_1 M is given. Let the set of states of M be S. We partition S into two disjoint subsets S^+ and S^- which corresponds to the sets of states when M is holding and not holding the marker in the finite control, respectively.[1] We assume that the initial state q_0 and the unique accepting state q_a of M are both in S^+. In order to make our proof clear, we also assume that M begins to move with its input head on the rightmost bottom boundary symbol # of an input tape and, when M accepts an input, it enters the accepting state at the rightmost bottom boundary symbol.

Suppose that an input tape x with $\ell_1(x)$=m and $\ell_2(x)$=n is given to M. For M and x, we define three types of mappings $f^{\uparrow -}{}_i : S^- \times \{0,1,\ldots,n+1\} \rightarrow S^- \times \{0,1,\ldots,n+1\} \cup \{\ell\}$, $f^{\uparrow +}{}_i : S^+ \times \{0,1,\ldots n+1\} \rightarrow S^+ \times \{0,1,\ldots,n+1\} \cup \{\ell\}$, and $f^{\downarrow -}{}_i : S^- \times \{0,1,\ldots,n+1\} \rightarrow S^- \times \{0,1,\ldots,n+1\} \cup \{\ell\}$ (i=0,1,...,m+1) as follows.

$f^{\uparrow -}{}_i(q^-,j) = \Big\{$ $(q^{-\prime},j')$: Suppose that we make M start from the configura-

----

1. Rigorously, S^- does not contain the states in which the input head of M positions on the same cell as where the marker is placed.

tion $(q^-,(i-1,j))$ with no marker on the input x (i.e., we take away the marker from the input tape by force). After that, if M reaches the i-th row of x in some time, the configuration corresponding to the first arrival is $(q^{-\prime},(i,j'))$;

Ω : Starting from the configuration $(q^-,(i-1,j))$ with no marker on the input tape, M never reaches the i-th row of x.

$f^{\uparrow +}{}_i(q^+,j)=$

$(q^{+\prime},j')$: Suppose that we make M start from the configuration $(q^+,(i-1,j))$. After that, if M reaches the i-th row of x with its marker held in the finite control in some time (so, when M puts down the marker on the way, it must return to this position again and pick up the marker), the configuration corresponding to the first arrival is $(q^{+\prime},(i,j'))$;

Ω : Starting from the configuration $(q^+,(i-1,j))$ with no marker on the tape, M never reaches the i-th row of x with its marker held in the finite control.

$f^{\downarrow -}{}_i(q^-,j)=$

$(q^{-\prime},j')$: Suppose that we make M start from the configuration $(q^-,(i+1,j))$ with no marker on the input tape (i.e., we take away the marker from the input tape by force). After that, if M reaches the i-th row of x in some time, the configura-

tion corresponding to the first arrival is (q^-
$'$,(i,j$'$)),

$\mathbb{Q}$: Starting from the configuration (q^-,(i+1,j)) with no marker on the tape, M never reaches the i-th row of x.

Below, we show that there exists a TR2-NTM(n log n) M such that T(M')=T(M). Roughly speaking, while scanning from the top row down to the bottom row of the input, M' guesses and checks $f^{\downarrow -}{}_i$, constructs $f^{\uparrow -}{}_i$ and $f^{\uparrow +}{}_i$, and finally at the bottom row of the input, M' decides by using $f^{\uparrow -}{}_{m+1}$ and $f^{\uparrow +}{}_{m+1}$ whether or not M accepts x (see Figure 1). In order to record these mappings for each i, O(n) blocks of O(log n) size suffice, so totally O(n log n) cells of the working tape suffice. More precisely, the working tape must be used as a "multi-track" tape. In the following discussion, we omit the detailed construction of the working tape of M'.

First, set $f^{\uparrow -}{}_0$, $f^{\uparrow +}{}_0$ to the fixed value $\mathbb{Q}$.
For i=0 to m+1, repeat the following. [$f^{\uparrow -}{}_i$, $f^{\uparrow +}{}_i$ are already computed at the (i-1)st row.]

(0) Go to the i-th row; When i=0, assume the boundary symbols on the first row.

(1) Guess $f^{\downarrow -}{}_i$; if i=m+1, set $f^{\downarrow -}{}_{m+1}$ to the fixed value $\mathbb{Q}$.

(2) [compute $f^{\uparrow -}{}_{i+1}$ from $f^{\uparrow -}{}_i$] When i≠m+1, do the following: Assume that there is no marker on the input tape. For each (q^-,j)∈$S^- \times$ {0,1,...,n+1}, start to simulate M from the configuration (q^-,(i,j)). While M moves only at the i-th row, behave just as M does. On the way of the simulation, if M would go up to the (i-1)st row at the k-th

column and would enter the internal state p^-, then search the table $f^{\uparrow -}_i$ to know the behavior of M above the i-th row. If the value $f^{\uparrow -}_i(p^-,k)$ is "Ω", write "Ω" into the block corresponding to $f^{\uparrow -}_{i+1}(q^-,j)$; If the value $f^{\uparrow -}_i(p^-,k)$ is "$(p^{-\prime},k')$", restart the simulation of M from the configuration $(p^{-\prime},(i,k'))$. While continuing to move in this way, if M would go down to the (i+1)st row, then write the pair of the internal state and column number just after that movement into the block corresponding to $f^{\uparrow -}_{i+1}(q^-,j)$ of the working tape. If M never goes down to the (i+1)st row (including the case when M enters a loop), then write "Ω" into the correspondent block.

(3) [compute $f^{\uparrow +}_{i+1}$ from $f^{\uparrow -}_i, f^{\uparrow +}_i$, and $f^{\downarrow -}_i$] When $i \neq m+1$, do the following: For each $(q^+,j) \in S^+ \times \{0,1,\ldots,n+1\}$, starting from the configuration $(q^+,(i,j))$, simulate M until M goes down to the (i+1)st row with the marker in the finite control. On the way of the simulation, if M would go up to the (i-1)st row with the marker held, then search the table $f^{\uparrow +}_i$ to know the behavior of M above the i-th row. If this value of $f^{\uparrow +}_i$ is "Ω", write "Ω" into the block corresponding to $f^{\uparrow +}_{i+1}(q^-,j)$; otherwise, restart the simulation of M from the configuration on the i-th row determined by the table value. If M puts the marker down on the i-th row of the input tape, then record the column number of this position in some track of the working tape and start the simulation of M which has no marker in the finite control. After that, If M would go down to the (i+1)st row or would go up to the (i-1)st row, then search the respective table $f^{\downarrow -}_i$ or $f^{\uparrow -}_i$ to find the configuration in which M return to the i-th row again. (If M never returns to the i-th row, write "Ω" into the block corresponding to $f^{\uparrow +}_{i+1}(q^+,j)$). From this configuration, restart the simulation of M. After that, if M returns to

the position where M put down the marker previously and picks it up, then continue the simulation of M; otherwise write " Q " into the block corresponding to (q^+,j). At some point of the simulation, If M goes down to the (i+1)st row with the marker held in the finite control, write the pair of the internal state which M would enter just after that time and the row number of this head position into the block corresponding to $f^{\uparrow +}_{i+1}(q^+,j)$. If M never goes down to the (i+1)st row, then write " Q " into the correspondent block.

(4) [check $f^{\downarrow -}_{i-1}$ from $f^{\downarrow -}_{i}$] When $i \neq 0$, do the following: In order to check that the table $f^{\downarrow -}_{i-1}$ guessed on the previous row is consistent with the table $f^{\downarrow -}_{i}$ (guessed at the present row), first newly compute a mapping $\underline{f}^{\downarrow -}_{i-1}$, which is uniquely determined from $f^{\downarrow -}_{i}$ and the content of the i-th row of the input. After this computation, check that $\underline{f}^{\downarrow -}_{i-1}$ is identical to the mapping $f^{\downarrow -}_{i-1}$ guessed at the previous row. If the equality holds, then continue the process; otherwise, reject and halt.

After the above procedure, on the (m+1)st row, M' begins to simulate M from the initial configuration $(q^+_0,(m+1,n+1))$ to decide whether or not M accepts the input after all. When M goes up to the m-th row with or without the marker, we can know how M returns again to the (m+1)st row, from $f^{\uparrow +}_{m+1}$ or $f^{\uparrow +}_{m+1}$, respectively. If M never returns to the (m+1)st row again, then M' rejects and halts. If M returns to the (m+1)th row, then M' continues the simulation. M' accepts the input x only if M' finds that M enters the accepting configuration $(q^+_a,(m+1,n+1))$.

It will be obvious that T(M)=T(M').

■

From Theorem 1 and Theorem 2, we get the following.

Corollary 1. $\mathcal{L}[2\text{-}DM_1] \subseteq \mathcal{L}[TR2\text{-}DTM(2^{O(n \log n)})]$.

We next investigate sufficient space for TR2-NTM's to simulate 2-NM_1. By using the same idea as in the proof of Theorem 2, we can show that the following theorem holds.

Theorem 3. $\mathcal{L}[2\text{-}NM_1] \subseteq \mathcal{L}[TR2\text{-}NTM(n^2)]$.

■

From Theorem 1 and Theorem 3, we get the following.

Corollary 2. $\mathcal{L}[2\text{-}NM_1] \subseteq \mathcal{L}[TR2\text{-}DTM(2^{O(n^2)})]$.

3. Necessary space.

In this section, we show that the algorithms described in the previous section are optimal in some sense. That is, those spaces are required for three-way Turing machines when the spaces depend only on one variable n (i.e., the number of columns of the input tapes).

Lemma 1. Let $T_1=\{x\in \{0,1\}^{(2)} \mid \exists n\geq 1[\ell_2(x)=n$ & (each row of x contains exactly one "1") & $\exists k\geq 2$[(x has k n-blocks) & (the last n-block is equal to some other n-block)]]$\}$. Then,

(1) $T_1\in \mathcal{L}[2\text{-}DM_1]$ and

(2) $T_1\notin \mathcal{L}[TR2\text{-}DTM(2^{L(n)})]$ (so, $T_1\notin \mathcal{L}[TR2\text{-}NTM(L(n))]$) for any $L:N\rightarrow R$ such that $\lim_{n\to\infty}[L(n)/n \log n]=0$.

Proof. (1): We can easily construct a 2-DM_1 M accepting T_1 as shown in Fig.2.

(2): The proof of Part (2) is lengthy, so ommited here.

■

Lemma 2. Let $T_2=\{x\in\{0,1\}^{(2)} \mid \exists n\geq 1[\ell_2(x)=n \ \&\ \exists k\geq 2[(x \text{ has } k \text{ n-blocks}) \ \&\ (\text{the last n-block is equal to some other n-block})]]\}$. Then,

(1) $T_2\in \mathcal{L}[2\text{-}NM_1]$,

(2) $T_2\notin \mathcal{L}[TR2\text{-}DTM(2^{L(n)})]$ (so, $T_2\notin \mathcal{L}[TR2\text{-}NTM(L(n))]$) for any $L:N\rightarrow R$ such that $\lim_{n\to\infty}[L(n)/n^2]=0$.

Proof. It is shown in [3] that Part (1) holds. From the same reason as in the proof of Lemma 1(2), we ommit the proof of Part (2).

■

From Lemma 1 and Lemma 2, we can conclude as follows.

Theorem 4. *To simulate 2-DM$_1$'s, (1) TR2-NTM's require Ω (n log n) space and (2) TR2-DTM's require $2^{\Omega(n \log n)}$ space in general.*

Theorem 5. *To simulate 2-NM$_1$'s, (1) TR2-NTM's require Ω (n^2) space and (2) TR2-DTM's require $2^{\Omega(n^2)}$ space in general.*

References

[1] A.Rosenfeld, *Picture Language*, Chapter 7 (Academic Press, NY, 1979).

[2] K.Inoue and I.Takanami, A Note on Deterministic Three-Way Tape-Bounded Two-Dimensional Turing Machines. *Inform. Sci.* **20**, pp.41-55 (1980).

[3] K.Inoue and A.Nakamura, Some Properties of Two-Dimensional Nondeterministic Finite Automata and Parallel Sequential Array Acceptors, *Trans. IECE Japan* **Sec.D**, pp.990-997 (1977).

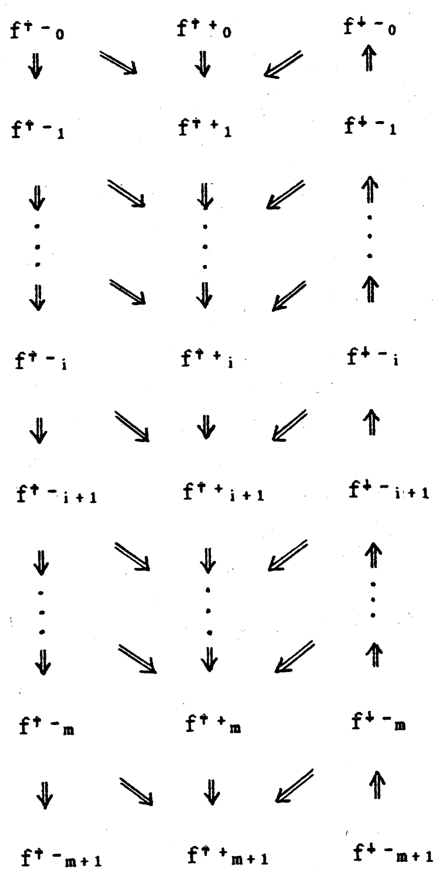

Fig.1. Mutual Dependences of the mappings.

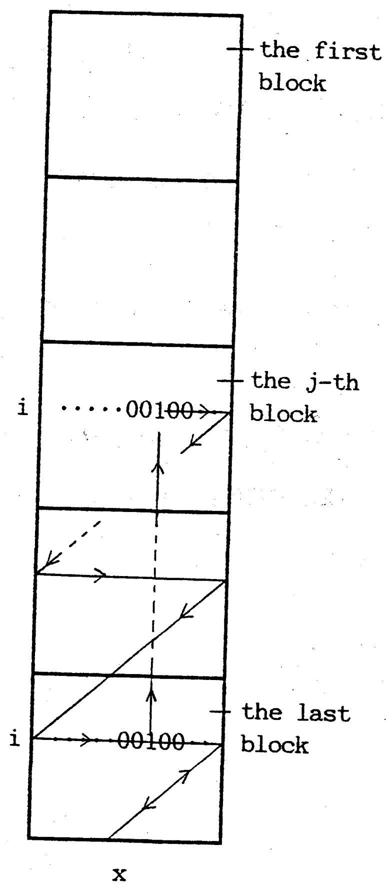

Fig.2. Action of 2-DM_1 M on a tape in T_1.